Nested Object Watermarking: From the Rectangular Constraint to Polygonal and Private Annotations

Claus Vielhauer
Otto-von-Guericke University Magdeburg
Universitätsplatz 2
39106 Magdeburg, Germany
+49-391-6718046
claus.vielhauer@iti.cs.uni-magdeburg.de

Maik Schott
Otto-von-Guericke University Magdeburg
Universitätsplatz 2
39106 Magdeburg, Germany
Tel+49-391-6718965
mschott@iti.cs.uni-magdeburg.de

ABSTRACT

A specific application for image watermarking is constituted by annotation watermarking (sometimes also called caption or illustration watermarking). In this domain, supplementary information is embedded directly in the media, so that additional information is intrinsically linked to media content and does not get separated from the media by non-malicious processing steps such as image cropping or compression. Recently, nested object annotation watermarking (NOAWM) has been introduced as a specialization in annotation watermarking, whereby hierarchical object relations are embedded in photographic images. Earlier work has suggested two methods for achieving this, whereby two main deficits can be identified for those: firstly both methods perform approximation of the user-generated, free-hand, polygonal shape annotations to rectangular annotation areas, which leads to a significant loss in precision of the annotation region after detection. Secondly, both of the previous algorithms do not introduce any security mechanisms, i.e. the object annotations are publicly accessible with no access control mechanism.

This paper presents an extension to one of the two previous approaches to suggest a first solution to these shortcomings. For this purpose, a novel method for embedding both the shape of the embedding region and the textual annotations in the phase coefficients of the Discrete-Fourier Transform (DFT) is presented. In order to achieve user dependent access control and improved synchronization for this scheme, the new method involves key-dependency. The key is used to provide an application level security feature and it is not designed yet in the sense of watermark key-space or sub-space security. The suggested methods have been prototypically implemented and first experimental results are presented in comparison to the original method with respect to transparency and robustness. We show that our new scheme may significantly reduce the transparency in terms of Peak-Signal-to-Noise Ratio (PSNR), whereas the overall watermark detection rate decreases, mainly due to local synchronization limitations.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MM&Sec'07, September 20–21, 2007, Dallas, Texas, USA.

Copyright 2007 ACM 978-1-59593-857-2/07/0009...$5.00.

Categories and Subject Descriptors

I.4.10 [**Image Representation**] Hierarchical, E.4 [**Coding and Information Theory**] Formal models of communication and K.6.5 [**Security and Protection**] Authentication.

General Terms

Algorithms, Management, Experimentation, Security

Keywords

Watermarking, Image Processing, Hierarchical Objects, Hierarchical Trees, Synchronization, Exhaustive Search, Access Control

1. Introduction

Annotation watermarking denotes a specific application of watermarks, which embeds supplementary information directly in the media. This is done with the goal that additional information is intrinsically linked to media content and does not get separated from the media by non-malicious image processing steps such as image cropping or compression.

Early image annotation watermarking approaches include methods optimized towards robustness to analog-to-digital conversion [1] and non-robust interactive segmentation with hidden object-based annotations [2]. From these early research, a special application for annotation watermarking called illustration watermarking has emerged, allowing illustration of images in an user interactive manner by adding keywords directly in the image itself, whereby [3] address computer generated images and [4] consider natural images. More recently, Nested Object Annotation Watermarking (NOAWM) has been introduced as a specialized annotation watermarking domain, whereby hierarchical object information is embedded in photographic images.

In a first approach to NOAWM, the Hierarchical Graph Concept (HGC) has been the introduced to model object relations, which are defined by users during an editing process, into a hierarchical tree structure. From this tree structure, a code-book decomposition of the annotation tree is generated and embedded in the cover using a block-luminance algorithm [5]. This code-book based approach has been extended by an alternative, high-capacity embedding scheme in [6], along with an additional concept for embedding object hierarchies by inheritance of signal properties: the Dual-Domain DFT (DDD) scheme. The later paper [6] further presents a comparative study for the three concepts, based on experiments measuring transparency and robustness to cropping and JPEG compression. One of the major observations here is that

regarding capacity, the novel DDD scheme has shown a major advantage with respect to the reconstruction rate of class hierarchies, because only one 32x32 pixel block per class is sufficient for retrieval, whereas the two other schemes required larger embedding regions. However, [6] concludes with at least three areas for future work: Firstly, embedding protocols for non-rectangular (i.e. polygonal) shapes of objects appear more realistic for annotations than the limitation to rectangular areas, imposed by all three methods. Secondly, the need for approaches to anti-collision can be seen, because in some cases the embedding protocol would overwrite the watermark for a parent class by its child annotation or vice versa. Finally, for application level security to achieve access control and for transparency reasons, the watermarks should be made key dependent and distributed in a pseudo-random manner across the annotation region, rather than based on protocols with fixed layouts as introduced earlier in [6].

This paper addresses two out of these three problems in extension to the DDD approach: watermarking for nested objects in non-rectangular, polygonal image annotation areas and an access control feature based on a key-controlled presence marker embedding. These novel schemes are evaluated by the same test methodology and test set introduced in [6] in comparison to the original DDD method. Evaluation goals are to observe the impact of polygonal embedding to transparency and robustness for the non-malicious image modification of JPEG compression.

This paper is organized as follows: after a short introduction to annotation watermarking of hierarchical objects in section 2, we expand upon our novel scheme for embedding (section 3) and retrieval (section 4) in polygonal regions. Section 5 summarizes the method for signal-intrinsic hierarchy embedding suggested in [6] and section 6 provides an experimental comparison to the later. Finally the paper is summarized in section 7 and future work is suggested.

2. Hierarchical Object Annotations

The concept of annotation watermarking requires user annotation of regions in a photograph, i.e. in our case spatial definition of objects and their semantic description, manually by users. For this purpose, our prototypical implementation supports pen-based displays, where users can directly trace the contours for spatial annotation. For assignment of visual-functional object relations to the annotation regions, a WordNet ontological browser is included in the system, for further details on WordNet see [7], for details of our prototype, see [5]. Figure 1 shows an example for polygonal annotations of objects “rock” and “lamp”, together with the related entries in the WordNet browser. This exemplary annotation leads to an object hierarchy, which can be represented by hierarchical graphs, as illustrated in Figure 2. In this graph representation, which has been suggested in [5], functional relations are modeled in a root-to-leave order. For example, the object ‘lamp’ consists of objects ‘pillar’ and ‘bulb’, thus leads to the simple graph shown in the figure.

Despite the possibility of polygonal annotations by users, the previously presented approaches for nested object watermarking have transformed these spatial annotations into rectangular regions before embedding the watermarks. This design has been chosen in the initial work for simplification of the embedding protocol. However, this leads to two obvious deficits of this concept: loss of accuracy in the spatial annotations and potential capacity loss due to embedding in additional regions (waste of potential watermarking areas), not belonging to the original objects.

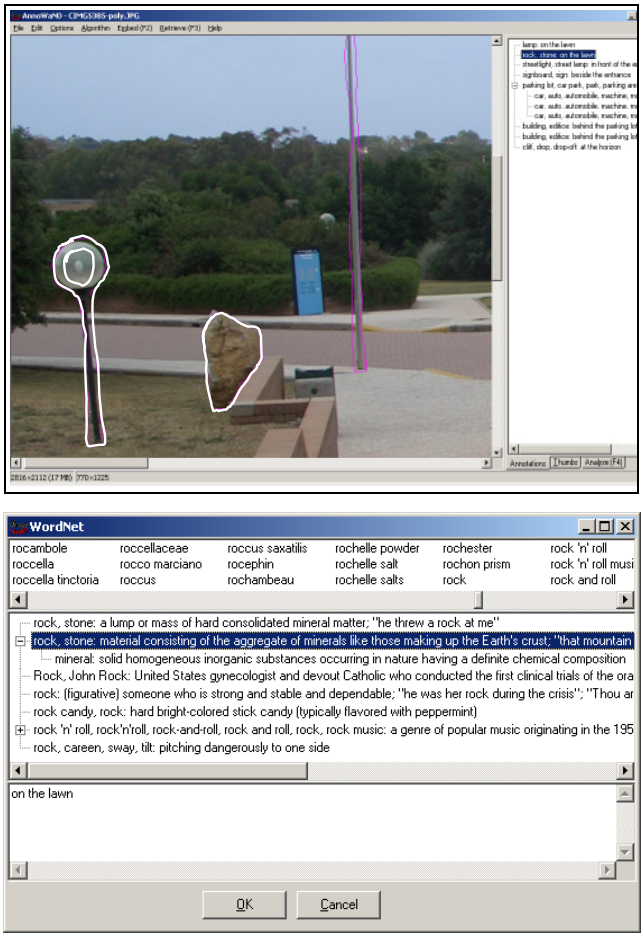

Figure 1. Example for a single hierarchy, polygonal annotation region (upper image) and its semantic annotation using WordNet (lower).

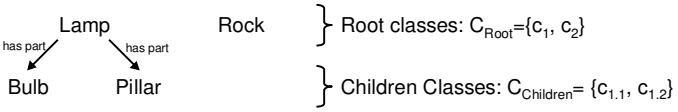

Figure 2. Object hierarchy for the exemplary annotations in Figure 1.

3. Embedding in Polygonal-Shaped Regions

The design principle of the original DDD algorithm from [6] is to separate the annotation data into two types of information, which are to be embedded in identical areas of photographs. The first type is the descriptive data (typically text), which we also refer to as the *message*, whereas the second type are the *hierarchical relations* between the objects. In DDD, this is accomplished by using the phase and magnitude components of a DFT (Discrete

Fourier Transformation), hence the name Dual-Domain DFT. Here, the message is embedded by modulating the phase, whereas the relation is embedded by modulation of magnitude in predefined frequency bands. Since phase modulation is considered to be more robust towards noise [8], it is used for message embedding. This is because the hierarchical relations between the watermarks in most cases – depending on the underlying model – are much more compact than the messages themselves, therefore they can be embedded with high redundancy in the magnitude part, compensating the lower robustness.

The DDD algorithm operates block-based, on blocks of luminances of block size $bx \times by$, whereby we have selected $bx=by=16$ for our implementation, in order to ensure sufficient number of frequency bands for embedding, as described in the following.

The DF-transformed luminance blocks are divided in a total number of bc frequency sub-bands of equal size, ranging from a lower cutoff frequency f_{cutoff} to the Nyquist frequency f_{nyq} (representing the highest frequency in each of the blocks). Because of the negative symmetry of the DFT, f_{nyq} is defined as $f_{nyq} = bx*by/2$. Two more system parameters, bc and f_{cutoff} control the number and width of the frequency bands used for embedding. In our implementation we have set them intuitively to $bc=5$ and $f_{cutoff} = 0.75 f_{nyq}$, we thus embed in 5 sub-bands in the highest 25% of the frequencies in the spectrum of each block after DFT.

Each of the embedding frequency sub-bands has therefore identical bandwidth: $bw = \lfloor (f_{nyq} - f_{cutoff}) / bc \rfloor$. For each of the sub-bands, indexed by b; $b \in \{1,\ldots, bc\}$, the upper frequency is defined as $f_{u,b}=(f_{nyq} - 1) - (bc-b)*bw$ and the lower one as $f_{l,b} = f_{u,b} - bw + 1$.

3.1 Message Embedding

The blocks used for embedding depend on the area, which has been selected for the object annotation by a user. In previous work, these user-specific object contours of arbitrary shape have been interpolated into rectangular bounding boxes, whereas the goal of the work presented in this paper is not to remain restricted to a rectangular embedding region, but support planes of arbitrary polygonal shape. In order to transform the polygonal user annotations, which are represented by sequences of pixel coordinates of the image into block coordinates, we apply the following new method:

For a given polygonal annotation, the entire image is partitioned into blocks of $bx \times by$ size, starting from the origin ($x=0$, $y=0$). Then for each of the resulting blocks, a decision whether or not it becomes an embedding block is taken based on the following rule: If at least $n_{vertices}$ vertices of this block are inside any of the selected polygonal object regions, this block is used for embedding. Here $n_{vertices}$ is a system parameter in the range of $[1, \ldots, 4]$, whereby lower values yield spatially more extensive regions for approximating the polygonal regions than higher values. As a result from this process, for each of the objects, we yield a polygonal shaped block-mask.

Figure 3 shows the polygonal shape for two different objects (light grey), as well as the resulting block masks for the cases where at least $n_{vertices}=3$ vertices of each block must lie inside the polygon (blocks marked with an x) and where only $n_{vertices}=1$ vertex were to be used (see empty blocks in the figure). As can be seen from the figure the blocks marks tend towards large spatial areas for lower $n_{vertices}=1$. For our further study, we consider $n_{vertices}=3$ for the generation of block masks, in order to keep the embedding blocks within close vicinity of the annotated object, thus avoiding the tendency to modify too many blocks only partly belonging to the actual annotation.

After determination of the block-wise embedding regions for each of the objects, three kinds of information are processed: i) a binary message m is generated from the textual annotations; ii) a so-called presence bit, indicating if a specific block belongs to a certain annotation or not and iii) object hierarchy information. Since the methods for the first two aspects have been developed significantly further from the origin of the DDD scheme in [6], we discuss these in more detail in the remaining parts of this section, whereas the hierarchy coding scheme is summarized briefly in section 5.

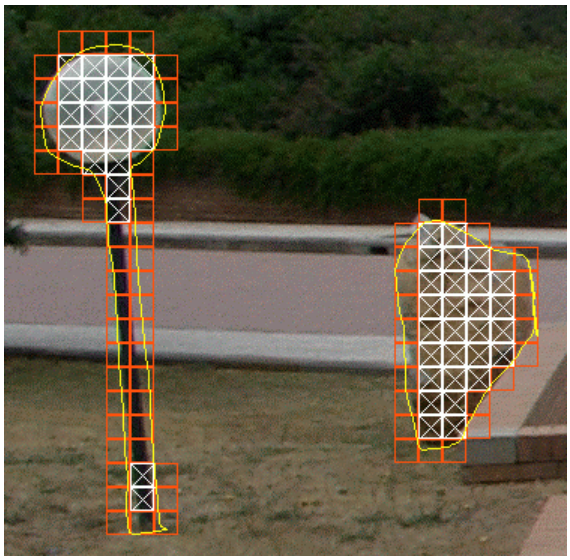

Figure 3. Two polygons (light grey) and their blocks for the 3 vertices criterion (blocks with an x) and a 1 vertex criterion (empty blocks).

For proper retrieval, the message m is coded by preceding its own message length in terms of number of bits $|m|$ to the actual message in the protocol. To further secure the embedded message (and its message length) against decoding errors due to noise, a Reed-Solomon (RS) Error Correction Code (ECC) is applied, where we have selected a (15, 9) code due to their encoding of 4 bit symbols, which is in line with our capacity of 4 bits per block. This ECC thus allows compensating up to 3 damaged blocks out of 15 blocks. The resulting length- and error-correction coded message is denoted as m', having a length in terms of number of bits $|m'|$.

The first block used for embedding m' and the presence bit for each of the objects is the upper leftmost block and the last the lower rightmost of each polygonal block mask; the block sequence is column-wise left to right and then row-wise top to bottom. Message m' is embedded in the frequency sub-bands $b \in \{1,\ldots, bc-1\}$, leaving out the sub-band $b=bc$. Consequently, m' is expanding over the first $\lceil |m'|/(bc-1) \rceil$ blocks. Additionally for every block a binary 1 is embedded in $b=bc$ to represent the shape

of the selected area in the sub-band of highest frequencies – the so-called presence bit. As outlined in the introduction to the DDD scheme, both *m'* and presence bit are embedded in the phase of the DFT domain, as detailed in the following subsection.

3.2 Phase rotation

Obviously, the embedded message bits can only be retrieved, if the bits are read in the same order. Because of the embedding scheme (embedding from the top leftmost to the bottom rightmost block) of the selected area and the detection of the selected area by the presence bits, the order detection may seem trivial at first sight. However, this is not the case, as the border between two objects with neighboring blocks could not be detected, since it would appear as a single one. Furthermore, the hierarchical concept implies that child watermarks may lay within their parent watermarks, thus blending with them. To overcome this, our scheme involves phase modulation in dependence of an output of a key-controlled Pseudo-Random Number Generator (PRNG), whereby in every block and in every sub-band, the bits are slightly differently embedded. For the sake of clarity, the magnitude of a frequency f within one block shall be denoted by $M(f)$, the phase by $\Theta(f)$.

Before the embedding of the first bit of *m*' of each object, all objects of an image are enumerated, whereby each object obtains a unique watermark index *wi*. Further, the PRNG is reset to the initial state defined by a user key and the index of the watermark. Recalling that the algorithm embeds block-wise $bc-1$ bits of *m*' in the sub-bands $\{1, \ldots, bc-1\}$ and the presence bit in the last sub-band *bc*, the actual phase embedding is done in the following manner*:* Given a single bit *bit* should be embedded into a sub-band *b*. If $bit=0$ then for each $f_{l,b} \le f \le f_{u,b}$, $\Theta(f)$ is set to $-0.5\pi + rnd$, otherwise to $+0.5\pi + rnd$, where *rnd* is a real number output of the PRNG in a certain state, having value in the range $[-\pi, \ldots, +\pi]$.

With an ideal PRNG, no phase of any block's sub-band frequencies would occur more than once. Therefore also the exact values of the first block's last sub-band with its presence bit is unlikely to occur at another place during the embedding of *m'* as the watermark, thus marking the beginning of the watermark. Due to the synchronization of the watermark sequence by a key- and object dependent PRNG, neighboring watermarks become distinguishable from each other, which would not be the case without it.

This embedding protocol works for any child objects having regions enclosed or partly enclosed by their parent objects, since blocks of children are omitted during embedding of the parent's watermark. The protocol in this case continues with the next not already watermarked block in embedding sequence, thus avoiding watermark collisions.

3.3 Magnitude Adaptation For Robustness

Although in the principal design of the DDD algorithm, the magnitudes are reserved for the hierarchy of the objects, there are cases, where it is advantageous to change them as well for the phase modulation during embedding. This is because the tendency of more robust phases in frequency bands of higher magnitudes. As a consequence for example, if a bit is embedded in a uniform region, where high frequencies do not exist (low magnitudes), phase modulation may become entirely unretrievable. To overcome this, the following actions are taken in our algorithm: First the magnitude threshold *tm* is initialized with an intuitively chosen value of 10. Immediately after the current bit is embedded a retrieval is performed. If a bit error occurred the magnitudes of the sub-band of the current embedded bit are compared against *tm*. Any magnitude of a sub-band showing an intensity of less than *tm* is set to *tm*. Then a re-embedding as well as a re-retrieval with a doubled *tm* is performed. This whole process is repeated until either no bit error occurs or *tm* exceeds an intuitively chosen maximum value of 10,000 for the sake of transparency.

4. Retrieval of Polygonal-Shaped Annotations

In this section the retrieval algorithm for polygonal-shaped annotations is described, whereby the following sub-section 4.1 firstly discusses our statistical method for message and presence bit retrieval, assuming the knowledge of precise block boundaries. The synchronization problem of finding those boundaries is discussed in sub-section 4.2.

4.1 Message Bits

To counter the effects introduced by noise after the embedding, statistical measures are used for bit retrieval in the phase domain. In our approach, this is achieved by means of a so-called "soft bit", whereby in the first stage, not a "hard bit" (either 0 or 1) but a fuzzy real value, or "soft bit" sb, $0 \le sb \le 1$ is returned by the retriever. A value 0 here means that the retriever is absolutely sure to read an embedded binary 0, a value of 1 indicates absolute confidence for a 1. All values in-between indicate a certain degree of in-confidence, with a maximum of 0.5, which signals that it is completely undecidable, if a binary 0 or 1 was embedded. In the following, our statistical retrieval method is described, whereby we focus on the retrieval of a single bit value *bit* (either presence bit or part of the embedded message *m'*).

From all the phases of the frequencies of the frequency band *b* associated to *bit*, the weighted mean μ and the weighted standard deviation σ are calculated. Since the higher the magnitude, the more robust is the phase against noise; the magnitudes in each sub-band are used as weights. If their sum:

$$m_{sum} = \sum_{f=f_{l,b}}^{f_{u,b}} M(f)$$

is zero, then the value for an undecidable state ($sb=0.5$) is returned. Otherwise, weighted mean μ and weighted standard deviation σ of the phases are calculated:

$$\mu = \frac{1}{m_{sum}} \sum_{f=f_{l,b}}^{f_{u,b}} \Theta(f) M(f)$$

$$\sigma = \sqrt{\frac{1}{m_{sum} - 1} \sum_{f=f_{l,b}}^{f_{u,b}} M(f)(\Theta(f) - \mu)^2}$$

Then, under the assumption of a normal distribution $N(\mu, \sigma^2)$, the probabilities for an embedded 0 or 1 are calculated: P_0 and P_1 respectively. The probability for an embedded 0 equals the probability of a given random phase X, satisfying both distribution parameters, to fall in the range of dispersion at -0.5π with a radius of 0.5π, where P is the cumulative probability density function for the normal distribution:

$P_0 = P(-\pi \le X \le 0)$ and analogously $P_1 = P(0 \le X \le +\pi)$, due to the symmetry and continuity of the normal distribution:

$$P(X \le x) = 1 - P(X > x) \quad \text{and}$$
$$P(x_1 \le X \le x_2) = P(X \le x_2) - P(X \le x_1)$$

for any $x_1 < x_2$. $P(X \le x)$ is calculated by the error function *erf* except in the case $\sigma=0$ where the Heavyside step function $H_{0.5}$ is used:

$$\mathrm{H}_a(b) = \begin{cases} 0 & \text{if } b < 0 \\ a & \text{if } b = 0 \\ 1 & \text{if } b > 0 \end{cases}$$

$$P(X \le x) = \begin{cases} H_{0.5}(x-\mu) & \text{if } \sigma = 0 \\ 0.5 + 0.5 erf\left(\dfrac{x-\mu}{\sigma\sqrt{2}}\right) & \text{otherwise} \end{cases}$$

The probabilities of an embedded 0 or 1 are then determined in relation to each other to get the "soft bit" *sb*:

$sb = (-P_0 + P_1 + 1) / 2$

For the final "hard bit" *bit* a threshold value $sbt \ge 0.5$ is introduced as system parameter, controlling the width of the exclusion area between the areas for an embedded 0 and 1.

$$bit = \begin{cases} 0 & if\ sb < 1 - sbt \\ 1 & if\ sb > sbt \\ Error & otherwise \end{cases}$$

The value of *sbt* depends on the sub-band. On the one hand, because the data of the sub-bands except the last are encoded with an RS ECC, an additional exclusion area is not needed, thus $sbt=0.5$. On the other hand, the presence bits have no such ECC, therefore needing an exclusion area, which we have intuitively set to $sbt=0.9$.

4.2 Synchronization

To detect a watermark, the PRNG is reset to the user-defined key and the watermarking object index *wi*, starting from 0. Over all blocks of the image, one block with a presence bit set is exhaustively searched. If none is found the watermark *wi* is assumed to be missing. If one is found, it is a candidate for the first block of an object annotation.

The problem in finding succeeding blocks for each candidate block is the complexity of probing each successor block in the remaining image as a potential successor and then re-applying the synchronization scheme recursively for this one as a new candidate. Since such exhaustive search is practically infeasible, for the sake of performance, the search space for subsequent block is limited to the block row of the actual candidate and the following row. Here, the subsequent block is searched for from all blocks starting from the found one till the last block of the next row. If none is found, it is assumed to be only this subsequent block, which is missing and this block is omitted. Otherwise, the subsequent block is the one, whose column distance to any already found preceding block is minimal. For this newly detected subsequent block, the same search method is applied in an iterative scheme, whereby the actually detected subsequent block becomes the actual one. This procedure terminates, if the found block either is the last block of the image or for the last $n_{row\text{-}length}$ blocks, no subsequent blocks were found, whereby $n_{row\text{-}length}$ denotes the number of blocks per row in the entire image. In the last case these row length blocks are removed from the found block list.

If the count of found blocks is not high enough to hold at least the parity bytes of an RS encoded message, the process is repeated at the next presence bit for the current *wi*. If the count of found blocks is higher or a next presence bit was not found, the whole process is repeated with an incremented *wi* at least 16 times since the last watermarks was found.

Note: This is a first intuitive approach to simplify the synchronization process. However, it may not provide optimal solutions.

5. Synopsis: Hierarchy Embedding

Embedding of the hierarchical relations of the annotated object has been completely adopted from the original scheme in [6], with the only difference, that the hierarchy, as well as message *m'* and presence bit, are embedded only in the polygonal-shaped region of each object, as described in section 3.1.

The general idea is based on inheritance of signal intrinsic properties to represent the hierarchical object relations. To do so, first a so-called hierarchical path from every actual object is generated. This is done by enumerating the nodes of the object hierarchy tree (see section 2) from the actual class node towards the root starting with zero and results in an ordered array of integers, representing the path to the root object for every annotated image object, up to a maximum path length (denoted as *d*).

Secondly, the algorithm unambiguously assigns one of the frequencies $f \in \{f_{cutoff}, \ldots, f_{nyq}\}$ to each object and constructs the hierarchy path of each object into an ordered list of representative frequencies, denoted as *hier*. Finally, for each object, the frequency representing the object itself receives a magnitude of the initial gain factor *s* multiplied with the maximum magnitude over all observed magnitudes in $\{f_{cutoff}, \ldots, f_{nyq}\}$ and deals as a coefficient to the magnitude. Each of the parent frequencies receives a relative degradation of the gain factor by $1/(d \bullet s^{1/2})$ for each node of the hierarchy path towards the root. Figure 4 illustrates this concept by plotting the magnitudes of a block after embedding for an exemplary annotation for the "lamp" example given in Figure 1, represented by $hier=(hier[0], hier[1])$, $s=20$ and $d=2$.

During retrieval, the method first performs a search for the frequency of highest magnitude in the embedding band and then reconstructs the hierarchy path in an iterative manner and in knowledge of the gain factor relation. A detailed description of this is given in the original article [6].

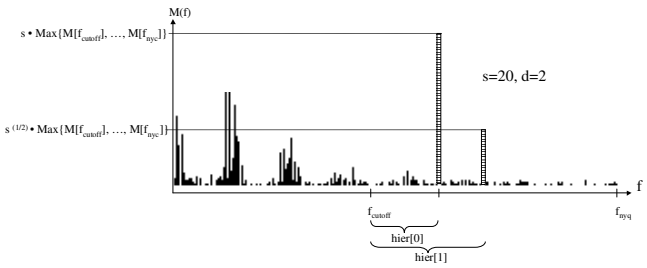

Figure 4. Example for the hierarchy embedding.

6. EXPERIMENTAL EVALUATION

The goal of the evaluation is to show the impact of an annotation based on polygons rather than rectangles to the transparency. To annotate objects based on rectangles, their shapes are simply modeled as bounding boxes. Therefore embedding is potentially done on many unnecessary blocks. As block-based polygons do not show this waste, a better transparency is expected.

The test set used for comparison is the same as Test Set B, introduced in [6] and consists of 15 images with at least 6 Mpixels size, having individual annotations, see detailed information in [6]. After the embedding, the images are altered by JPEG compression with different qualities. As shown in the previous paper [6], the DDD algorithm was only robust to certain degree up to JPEG quality level 75. Therefore less quality levels are omitted from the comparison in Table 1 * MERGEFORMAT . "Raw" denotes no compression at all and J<x> denotes a JPEG compression with a quality level of x. In Table 1 * MERGEFORMAT , the detection rates are shown separately for the first synchronization blocks (first row, Sync. Detection Rate) and the success rates for retrieval of the entire watermark (second row, Watermark Detection Rate) to measure a potential loss due to erroneous synchronization of subsequent blocks.

	Original DDD				Polygonal DDD			
Compression	**Raw**	**J100**	**J90**	**J75**	**Raw**	**J100**	**J90**	**J75**
Sync Detection Rate [%]	98.59	98.59	20.42	0.00	83.26	81.34	5.43	0.00
Watermark Detection Rate [%]	98.59	98.59	20.42	0.00	70.04	66.41	2.07	0.00
Average PSNR [dB]	45.93	45.43	40.42	39.00	49.04	48.13	41.35	39.74
Minimum PSNR [dB]	38.61	38.52	36.52	34.83	42.05	41.88	38.27	35.66
Maximum PSNR [dB]	52.97	51.74	46.15	43.90	55.94	53.77	46.81	47.19
Standard Deviation [dB]	4.18	3.86	2.23	2.54	4.24	3.68	2.16	3.20

Table 1: Synchronization & Watermark detection rate, average, minimum, maximum and standard deviation of PSNR at 3 different compression rates.

As initially expected, we observe that the average transparency improved by 3.11 dB for the case of raw images and between 2.70 and 0.74 dB for the three JPEG compression levels.

However, the new DDD algorithm is less robust than the original DDD algorithm. We assume that the robustness loss has two main causes: firstly, the original algorithm used rectangular watermarks and therefore it was only necessary to properly retrieve the height and width of the watermark for synchronization. The new algorithm supporting polygons has no knowledge about the shape and cannot use such a-priori knowledge during the retrieval. If a block was not found it is therefore very hard to decide if this block was never marked or if the marks were removed by noise (from JPEG compression, etc.). Secondly, the current watermarking synchronization algorithm is a first intuitive approach and work in progress. For example, one idea is selecting all potential next blocks for presence bit detection instead of selecting only one next block in the synchronization step during presence bit retrieval.

Note that because of the block-based nature of the algorithm and its operation in frequency-domain, no robustness against geometric operations is expected. Therefore such evaluation was omitted. However due to the local synchronization scheme, our method has shown robustness against cropping, provided that all watermarked blocks are preserved in the cropping. The latter would not be the case for example in a binary masking scheme, whereby a compressed version of a binary mask for all objects is embedded across the entire image, which would obviously very likely be destroyed by arbitrary cropping operations.

7. SUMMARY AND FUTURE WORK

In this paper, we have presented an algorithm extension to an earlier method for hierarchical object annotations in photographic images. Within our experimental tests, the enhanced method shows the following main advances and disadvantages (with respect to transparency and robustness as well in its overall functionality) compared to the original algorithm as follows:

Firstly, whilst the original method only supports rectangular areas, which is somewhat contradictory to the intuition of pen-based annotation watermarks, the extension suggests a scheme for approximating polygonal-shaped embedding regions. Secondly, a security-enhancing effect is introduced to prevent unauthorized retrieval of textual annotations using user specific keys. This may allow future applications of nested object watermarking to scale the access such that for example the object hierarchy information part remains public, while the spatial and textual information remains disclosed to holders of legitimization keys. Finally, the overall transparency of the scheme could be improved by reducing the proportion of blocks modified for embedding and the entire number of blocks in the image. In particular, our experimental evaluation has shown that the average PSNR has improved from 45.93 dB to 49.04 dB for the same uncompressed test data as in the initial evaluation.

However, while the detection of the presence bits, representing the polygonal shape of annotations, has shown sufficient robustness in our evaluation, we have observed problems in the robustness of textual annotation messages. In our experiments, this has resulted in a significantly less robust watermark detection rate of 70.04% in comparison to the original (98.59%) for raw images.

Apparently, this is partly a synchronization problem due to the chosen retrieval protocol, which pursues a limited local search strategy adopted from the retrieval of rectangular areas. This conclusion is supported by the observation that the detection of the first synchronization block is more reliable (83.26% for uncompressed images) than retrieval of the entire watermark. Here, future work will include alternate search strategies and the analysis of image content features, such as object contours, to enable improved local synchronization strategies. Another reason for this problem could be a current sub-optimal configuration of the RS ECC parameters, which provides limited intra-block error correction for groups of 15 blocks. Additional future work will address the improvement of subjective transparency. Due to the current strategy, to embed information continuously in neighboring blocks, the subjective transparency can be ranked at medium level. Finally, our aim is to study the possibility of utilizing content specific properties, such as visual hashes or histograms to inherit properties within nested object hierarchies.

8. ACKNOWLEDGMENTS

This work has been sponsored by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA8655-07-1-3013. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government. The pen-based interaction paradigm was motivated by our work in the EU Network of Excellence SIMILAR (Proposal Reference Number: FP6–507609). The contents of this publication are the sole responsibility of the authors and can in no way be taken to reflect the views of the European Union.

9. REFERENCES

[1] Perry, B. and MacIntosh, B., and Cushman D. *Digimarc MediaBridge: the birth of a consumer product from concept to commercial application*, Proc. SPIE Vol. 4675, p. 118-123, Security and Watermarking of Multimedia Contents IV, 2002

[2] Rytsar, Y. and Voloshynovskiy, S. and Ehrler, F. and Pun, T. *Interactive segmentation with hidden object based annotations: towards smart media*, In Proceedings of SPIE Electronic Imaging, Storage and Retrieval Methods and Applications for Multimedia, San Jose, USA, 2004

[3] Sonnet, H. and Isenberg, T. and Dittmann, J. and Strothotte, T. *Illustration Watermarks for Vector Graphics*, In Proceedings of IEEE Pacific Conference on Computer Graphics and Applications, pp. 73-82, 2003

[4] Vogel, T and Dittmann, J. *Illustration Watermarking: An Object Based Approach for Digital Images*. Proceedings of SPIE 2005, Jan. 20 - 27, San Jose, California, USA, 2005

[5] Vielhauer, C. and Schott, M. *Image Annotation Watermarking: Nested Object Embedding using Hypergraph Model*, Proceeding of the 8th ACM Workshop on Multimedia and Security, pp. 182 – 189, 2006

[6] Vielhauer, C. and Dittmann, J. *Nested Object Watermarking: Comparison of Block-Luminance and Blue Channel LSB Wet Paper Code Image Watermarking*, Proceedings of SPIE Electronic Imaging - Security, Steganography and Watermarking of Multimedia Contents IX, Vol. 6505, pp. 65050L, 2007

[7] Fellbaum, C. *WordNet: An Electronic Lexical Database (Language, Speech and Communication)*, MIT Press, ISBN 0-262-06197-X, 1998

[8] Hayes, M.H. *The reconstruction of a multidimensional sequence from the phase or magnitude of the FFT*. IEEE Transactions on Acoustics, Speech and Signal Processing, 1992